Quantum-Inspired Convolutional Reinforcement Learning for Interpretable and Optimized Multi-Stock Trading Investment Strategies and Portfolio Management

Dr. Devadutta Indoria [1]*, Prof. Bhuvan Unhelkar [2], Prof Siva Shanker S [3]
[1]Postdoc Research Scholar, Muma College of Business, University of South Florida, Tampa, Florida, USA
[2]Professor, Muma College of Business, University of South Florida, Tampa, Florida, USA
[3]KGRCET, Hyderabad, India
Email: [1]mailmedevdutt@gmail.com; [2]bunhelkar@usf.edu; [3]drsivashankars@gmail.com
Orcid: [1]https://orcid.org/0000-0002-4556-9458; [2]https://orcid.org/0000-0003-1118-3837; [3]https://orcid.org/0000-0002-7616-6088

*Corresponding Author: Dr. Devadutta Indoria

Abstract
Traditional financial models often rely solely on historical price data, failing to account for real-time market complexity and sentiment. This research addresses this gap by proposing a quantum-inspired reinforcement learning framework that integrates sentiment indicators and market conditions for optimized multi-stock trading decisions. A quantum-inspired financial modeling methodology is used to maximize trading decisions, with Rewards and Sentiment indicators based on Convolutional Revolution Reinforcement Learning (RMS-CRRL) and Quantum Finance Theory (QFT). This research suggests a multi-stock trading strategy with Dow Jones 50 component stocks, where the financial information is obtained from the Yahoo Finance and Alpha Vantage APIs. The model leverages data preprocessing methods such as missing value interpolation, outlier detection, and consistency enhancements to provide high-quality data. The novelty of the model lies in its integration of market conditions and consumer sentiment, setting it apart from conventional models that rely solely on historical price data. The RMS-CRRL structure utilizes convolutional neural networks for stock price forecasting to minimize uncertainty in the financial market. The revised revolution optimization algorithm is applied to model stability and the best choice of hyperparameters. In addition, the interpretability of the convolutional neural networks-based predictions is guaranteed using rectangular constraints for local interpretable model-agnostic explanations providing explainability in trade decisions. Comparisons for performance show that the suggested model outperforms baseline methods such as traditional DDPG and A3C models, achieving cumulative returns of 22.4%, which reflects a 7.2% improvement in annual returns. The RMS-CRRL approach also demonstrates an average portfolio value benefit of 10-15% compared to the conventional approach.
Keywords: Multi-stock Trading Strategy, Dow Jones 50 Constituents, Financial Data Analysis, Quantum Finance, Portfolio Management, Stock Price Prediction, Trading Decision Optimization, Performance Comparison.

1. Introduction

In the constantly changing environment of international finance, volatility and complexity of stock markets pose ongoing threats to investors and financial institutions alike [1-2]. Conventional investment schemes, based on statistical models and past trends, are increasingly supplemented by data-driven strategies fueled by Artificial Intelligence (AI) [3]. As financial information becomes more multidimensional, granular, and high-frequency, there is increasing demand for smarter systems that can process enormous amounts of data and make adaptive and informed decisions [4]. Such systems should earn profit and effectively handle risk in rapidly changing and uncertain market conditions [5]. Among the many AI methods, Reinforcement Learning (RL) is highly effective in decision-making in sequential action and delayed reward situations as applicable to financial trading [6]. This study notes that applying reinforcement learning to portfolio management involves key challenges such as the curse of dimensionality, sparse rewards, and the complexity of black-box models [7][8]. To overcome these limitations and provide greater explanations, hybrid models that combine RL with deep models such as Convolutional Neural Networks (CNNs) are used in recognizing spatial and temporal patterns in financial time-series data [9].

Combining deep learning, quantum-inspired computing, and reinforcement learning represents a watershed moment in the engineering of smart financial systems [10]. By providing a richer understanding of market behavior, these combined models open up the path for interpretable, scalable, and optimized multi-asset trading strategies [11]. This research develops and validates the combined models for a rapidly changing environment, resulting in a smart framework that can cope with the intricacies of contemporary financial markets.

The research is organized as follow: section 1 covers the introduction section, section 2 focuses on the literature review, section 3 focuses on the proposed methodology, section 4 focuses on the results and discussions and section 5 focuses on the conclusion and future scope.

2. Related Works

Zou, J., et al. [12] investigated the limitations of traditional Deep Reinforcement Learning (DRL) models in financial environments characterized by low signal-to-noise ratios and proposed a Cascaded LSTM-based PPO (CLSTM-PPO) framework. *Cheng, C., et al. [13]* investigated a multi-agent trading system called Quantum Finance-based Forecasting Reinforcement Learning (QF-FRL), which integrates quantum finance with fuzzy reinforcement learning to optimize stock trading strategies. The system employs TD3-based deep reinforcement learning for trade decisions and a fuzzy inference system for risk control, yielding superior returns and robust risk-adjusted performance compared to traditional models. *Ojha, N.K., et al. [14]* highlighted the transformative role of Artificial Intelligence (AI) in the financial sector, particularly within banking, where AI enhances commerce automation and predictive capabilities by leveraging historical user data. *Ansari, Y., et al. [15]* proposed a reinforcement learning-based multi-stock trading framework that integrates enhanced state representations from technical and fundamental indicators and a novel PSR reward function. *Sattar, A., et al. [16]* developed a sentiment-aware trading framework called RMS-Driven DRL, incorporating news sentiment and consumer data alongside earnings reports to inform portfolio decisions. The research evaluated

PPO_RMS, A2C_RMS, and DDPG_RMS models, with DDPG_RMS achieving a 27% return and a Sharpe ratio of 0.66, outperforming the Dow Jones index and demonstrating enhanced risk-return balance. *Cheng, L.C. et al. [17]* proposed a multi-agent adaptive trading framework utilizing reinforcement learning to address generalization issues in existing financial models. By incorporating the TD3 algorithm for long-short strategy execution and adaptive capital allocation, the system demonstrated improved stability and outperformed traditional single-agent RL models in portfolio management. *Gu, F., et al. [18]* introduced the MTS framework, a time-aware deep reinforcement learning model with integrated short-selling and risk control via Incremental Conditional Value at Risk. Utilizing an encoder-attention mechanism, MTS outperformed traditional and advanced ML models across five datasets, achieving up to 30.67% higher cumulative returns and enhanced risk-adjusted metrics such as the Sharpe and Sortino ratios. *Boonpan, S. et al. [19]* proposed deep neural network models integrating advanced data preprocessing techniques such as dollar bar sampling, trend scanning labeling, and piecewise aggregate approximation to predict directional returns of major US tech stocks. *Cortés, D.G., et al. [20]* developed an explainable reinforcement learning model for portfolio management within a simulated CAC-40 trading environment, aiming to address transparency challenges in financial machine learning. *Park, J.H., et al. [21]* implemented a Deep Reinforcement Learning (DRL) trading robot using Proximal Policy Optimization (PPO) to evaluate performance improvements. By comparing a robot trained with only Apple Inc. stock data to one trained with Apple, the S&P 500 index, and US interest rate data, the research highlights the effect of incorporating broader market factors on trading performance. *Garcia, R.D., et al. [22]* proposed a real-time decision-making support system for stock markets, utilizing complex networks and blockchain technology to reduce transaction costs and enhance transparency. By analyzing multi-source marketplace data through cloud computing, the system effectively simplifies market analysis and supports decentralized transactions, as shown in its evaluation using NASDAQ data. *Zhu, Q., et al. [23]* introduced the Dynamic Inductive Predictive Graph Neural Network (DIP-GNN), which combines sequence decomposition and dynamic evaluation methods like Pearson's correlation and dynamic time warping to improve short-term stock price forecasting. *Guan, B., et al. [24]* proposed the K-means-CAE-LSTM model, which integrates clustering, convolutional autoencoder-based image feature extraction, and long short-term memory networks to predict stock prices. *Shi, M., et al. [25]* introduced the STAGE framework, combining Graph Attention Network (GAT), Variational Autoencoder (VAE), and Sparse Spatiotemporal Convolutional Network (STCN) to improve stock prediction accuracy and anomaly data detection. *Luxenberg, E., et al. [26]* addressed the problem of portfolio optimization using cumulative prospect theory (CPT) utility, which is not concave in portfolio weights. Research demonstrated that system utility expressed as a difference of functions and derived a global lower bound for its maximization using a minorization–maximization (MM) algorithm.

2.1 Problem Statement

The complexity and volatility structure of financial markets are rapidly growing, which complicates the creation of strong and efficient stock trading strategies. Traditional portfolio management systems using xxx fail to follow fast-changing market dynamics, making investment

decisions worse and financial risks higher. Furthermore, the non-transparency of most trading models causes trust and interpretability concerns, particularly in high-risk financial settings. This research answers these challenges through a proposed general and adaptive framework of trading with the goal to enhance the accuracy of stock forecasting, facilitate decision-making under uncertainty, and provide interpretable clear insights to back sound and sustainable investment decisions.

2.2 Research Objectives
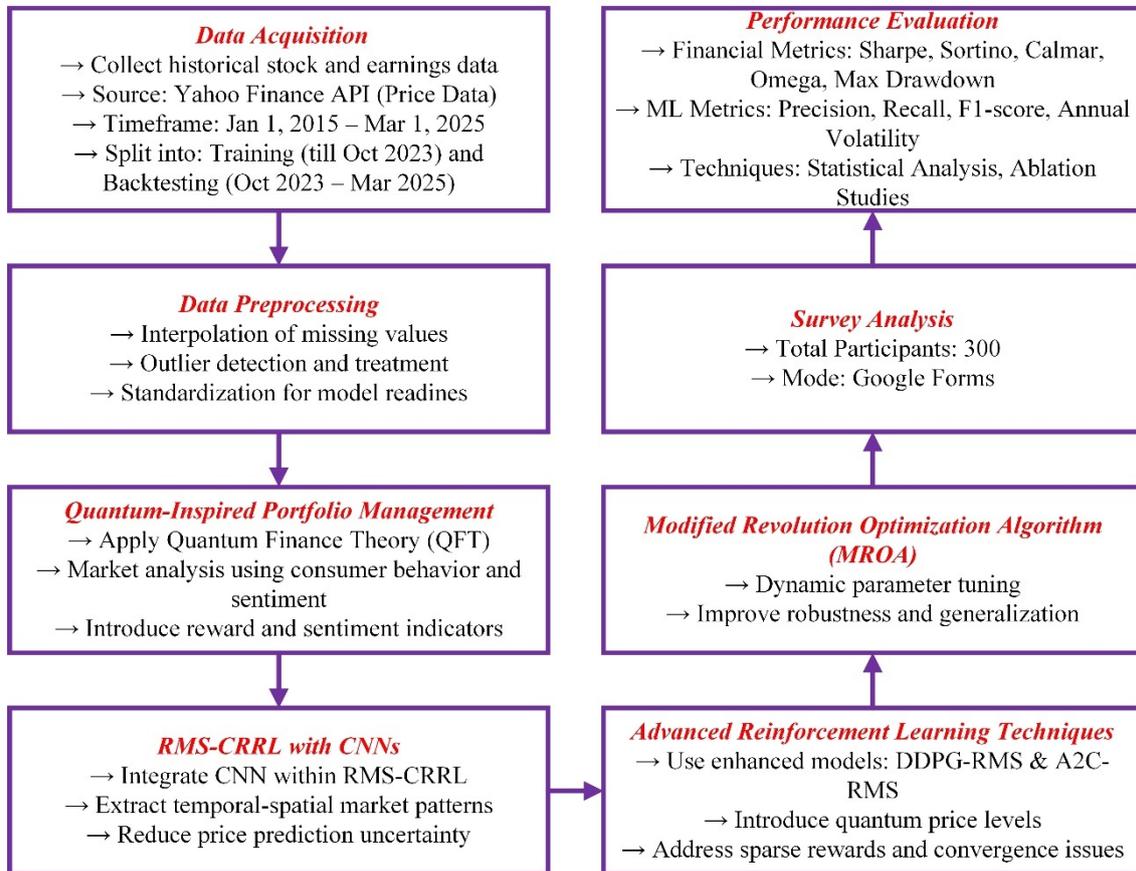*The research objectives of the works are as follow;*
- To develop a novel multi-stock trading strategy using deep reinforcement learning by integrating high-quality financial and sentiment data including earnings reports and market behavior insights from the Dow Jones 50 constituents, to overcome the limitations of single-source, price-only models and enable more adaptive, informed, and diversified trading decisions.
- To design a quantum-inspired portfolio management system that combines Convolutional Revolution Reinforcement Learning (RMS-CRRL) and Quantum Finance Theory (QFT), to enable risk-aware and sentiment-driven trading strategies that better capture market complexity and uncertainty compared to classical reinforcement learning methods.
- To enhance both prediction accuracy and interpretability by incorporating CNNs, improved Deep Deterministic Policy Gradient (DDPG), Advantage Actor-Critic (A2C), and Rectangular Constraints for Local Interpretable Model-Agnostic Explanations (R-LIME), to not only improve performance but also provide transparency and trust in AI-driven financial decision-making an often-overlooked need in algorithmic trading.
- To rigorously validate and benchmark the proposed framework using advanced evaluation metrics (e.g., Sharpe, Sortino, Omega ratios), ablation studies, and statistical analyses, to demonstrate measurable superiority over traditional reinforcement learning and trading models, ensuring practical relevance, scalability, and robustness of the approach.

3. Proposed Methodology for Quantum-Inspired Convolutional Reinforcement Learning for Interpretable and Optimized Multi-Stock Trading Investment Strategies
Figure 1 illustrates a quantum-inspired deep reinforcement learning framework for stock market prediction and portfolio management. It begins with Data Acquisition, where historical stock data (2015–2025) is collected from the Yahoo Finance Application programming interface (API) and split for training and back testing.

During the Data Preprocessing phase, the data is cleaned by interpolation, outlier handling, and standardization. Quantum-inspired portfolio management combines QFT with reinforcement learning to research market behavior based on sentiment and reward signals. Deep reinforcement learning techniques such as DDPG-RMS and A2C-RMS are utilized to enhance risk decision-making, manage sparse rewards, and implement quantum price levels. The Dynamic Hyperparameter Tuning is assisted by the Modified Revolution Optimization Algorithm (MROA) in making the model more robust and generalize well. Lastly, Performance Evaluation monitors

the strategy through financial metrics (Sharpe, Sortino, Calmar, etc.) and machine learning metrics (Precision, Recall, F1-score) assisted by statistical analysis and ablation studies.

Data Acquisition
→ Collect historical stock and earnings data
→ Source: Yahoo Finance API (Price Data)
→ Timeframe: Jan 1, 2015 – Mar 1, 2025
→ Split into: Training (till Oct 2023) and Backtesting (Oct 2023 – Mar 2025)

Data Preprocessing
→ Interpolation of missing values
→ Outlier detection and treatment
→ Standardization for model readines

Quantum-Inspired Portfolio Management
→ Apply Quantum Finance Theory (QFT)
→ Market analysis using consumer behavior and sentiment
→ Introduce reward and sentiment indicators

RMS-CRRL with CNNs
→ Integrate CNN within RMS-CRRL
→ Extract temporal-spatial market patterns
→ Reduce price prediction uncertainty

Performance Evaluation
→ Financial Metrics: Sharpe, Sortino, Calmar, Omega, Max Drawdown
→ ML Metrics: Precision, Recall, F1-score, Annual Volatility
→ Techniques: Statistical Analysis, Ablation Studies

Survey Analysis
→ Total Participants: 300
→ Mode: Google Forms

Modified Revolution Optimization Algorithm (MROA)
→ Dynamic parameter tuning
→ Improve robustness and generalization

Advanced Reinforcement Learning Techniques
→ Use enhanced models: DDPG-RMS & A2C-RMS
→ Introduce quantum price levels
→ Address sparse rewards and convergence issues

Figure 1: Architecture of proposed methodology for quantum-inspired convolutional reinforcement learning for interpretable and optimized multi-stock trading investment strategies.

3.1 Analysis of the Rewards and Sentiment indicators based on Convolutional Revolution Reinforcement Learning

The analysis of rewards and sentiment indicators using CRRL focuses on integrating market dynamics with real-time sentiment data to enhance decision-making in intelligent agents. CRRL employs convolutional layers to extract temporal and spatial features from sentiment-augmented states, enabling the agent to learn optimal policies [27-28]. Figure 2 depicts the architecture of convolutional revolution reinforcement learning for sentiment indicators.

System begins by representing the state as a fusion of market/environmental features and sentiment cues.

$$S_t = \phi(X_t, \sigma_t)$$

(1)

The state S_t is a nonlinear combination (feature encoding) of market data X_t and sentiment indicators σ_t, system need to take into account that $\sigma_t \in R^n$ is the sentiment vector (e.g. from NLP-processed tweets or news headlines).

To extract higher-order spatiotemporal patterns from state input, the state is passed through a 1D convolution layer to produce temporal patterns in sentiment-reinforced signals.

$$f_t = \mathrm{Re}\,LU(Conv1D(S_t;W_c)+b_c) \tag{2}$$

where $f_t \in R^k$ signifies the extracted deep features, $Conv1D$ signifies the 1D convolution operation, W_c defines the convolution kernel weights, b_c signifies the bias term and $\mathrm{Re}\,LU$ is the rectified linear unit Activation function.
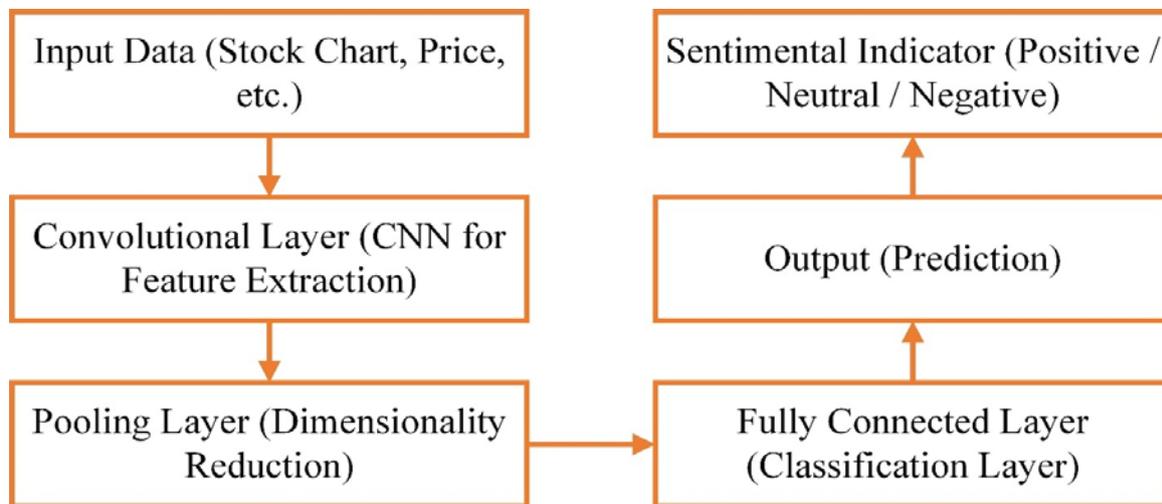
Figure 2: Architecture of convolutional revolution reinforcement learning for sentiment indicators

The value of being in state S_t is approximated using the extracted features, which help predict the expected return from that state.

$$V(s_t) = f_t^T W_v + b_v \tag{3}$$

where $V(s_t) \in R$ signifies the estimated value of state, W_v signifies the linear weight vector, b_v denotes the scalar bias.

Utilizing bellman expectation to estimate Q-value as immediate reward discounted future value.

$$Q(S_t,\alpha_t) = E_{st+1}[r_t + \gamma V(s_{t+1})] \tag{4}$$

Where $Q(S_t,\alpha_t)$ defines the expected return of taking α_t at state S_t, r_t signifies the reward received at time t, $\gamma \in [0,1]$ defines the discount factor for future rewards.

The reward is influenced both by environmental return and sentimental signal.

248

$$\gamma_t = \alpha.R_{env}(S_t, \alpha_t) + \beta.SentScore(\sigma_t) \tag{5}$$

Where R_{env} signifies the reward from environment, $SentScore(\sigma_t)$ defines the sentiment score (e.g. polarity and subjectivity), α signifies the weighting coefficient for blending.

To analyze the measure of prediction error between successive value estimates.

$$\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t) \tag{6}$$

Where $\delta_t \in R$ signifies the TD error used for learning updates.

For analyzing the value network loss function, loss used to value the network, minimized through backpropagation

$$L_v = \frac{1}{2}\delta_t^2 \tag{7}$$

For dynamically adjusts the sentiment weight β to improve reward correlation while reducing sentiment noise.

$$\beta_{t+1} = \beta_t + \eta.\nabla_\beta (E[r_t^2] - \lambda.Var(SentScore(\sigma_t))) \tag{8}$$

where β_t signifies the sensitivity weight, η defines the learning rate for meta-parameters, λ defines the regularization coefficient.

3.1.1 Analysis of the Quantum Finance Theory

Quantum finance theory applies principles of quantum mechanics to financial modeling, offering a novel approach to understanding market behavior and uncertainty [29]. It uses concepts like superposition and entanglement to model asset pricing and risk more accurately.

The financial market is modeled as a quantum system in a superposition of all possible market states $|s_i\rangle$, each weighted by a complex probability amplitude c_it.

$$|\psi(t)\rangle = \sum_{i=1}^{N} c_i(t)|s_i\rangle \tag{9}$$

where, $|\psi(t)\rangle$ signifies the quantum state of the market at time t, $|s_i\rangle$ signifies the basis states (e.g. bullish, bearish, neutral market), $c_i(t)$ defines the complex probability amplitudes, N defines the number of distinguishable market states.

The probability of the market being in state s_i is given by the squared modulus of its amplitude. This captures the inherently probabilistic nature of quantum finance.

$$P(S_i, t) = |C_i(t)|^2 \tag{10}$$

where $P(S_i, t)$ signifies the probability of market being in state s_i, $C_i(t)$ defines the born rule from quantum mechanics.

The evolution of the quantum financial system is governed by the Hamiltonian, $\hat{H}$ which encodes market forces such as volatility, interest rates, and trader behavior.

$$ih\frac{d}{dt}|\Psi(t)\rangle = \hat{H}|\Psi(t)\rangle$$
(11)

where i signifies the imaginary unit, h denotes the reduced Planck's constant (set to 1 in financial analogies), $\hat{H}$ signifies the market Hamiltonian operator, $\frac{d}{dt}|\Psi(t)\rangle$ defines the time derivative of the market state.

The expected value of a financial observable (e.g. asset price, portfolio return) is computed as the quantum expectation over the system state using a Hermitian operator $\hat{A}$.

$$\langle\hat{A}\rangle = \langle\psi(t)|\hat{A}|\psi(t)\rangle$$
(12)

where $\hat{A}$ signifies the observable operator (e.g. price, volatility), ψ signifies the quantum expectation value at time t.

The quantum variance of an observable defines financial risk in a non-classical way, accounting for probabilistic fluctuations inherent in the system.

$$Var(\hat{A})_t = \langle\Psi(t)|\hat{A}^2|\Psi(t)\rangle - (\langle\hat{A}\rangle)^2$$
(13)

Where $Var(\hat{A})_t$ defines the risk or uncertainty of observable $\hat{A}$, $\hat{A}^2$ defines the square of the operator (reflects nonlinearity in measurements).
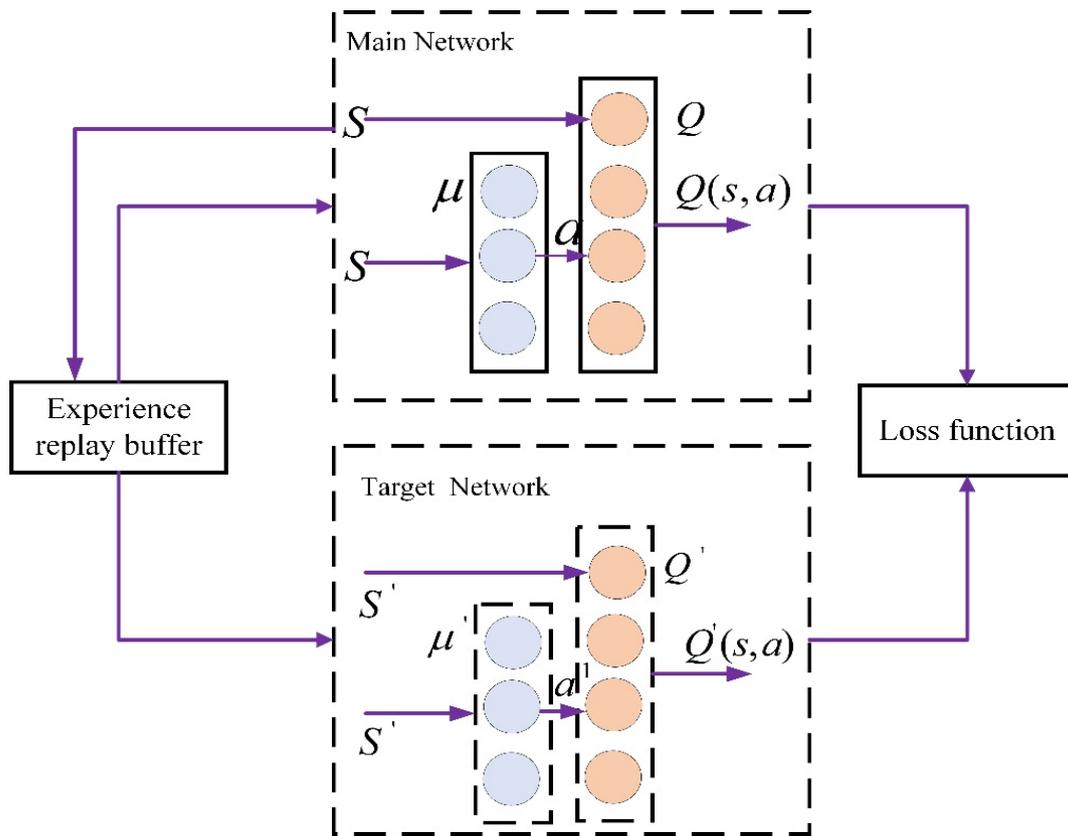
3.2 Analysis of Deep Deterministic Policy Gradient for the Decision Making

Deep deterministic policy gradient is a model-free, actor-critic reinforcement learning algorithm designed for continuous action spaces [30-31]. It combines deep neural networks with deterministic policy learning to optimize actions directly from high-dimensional input states. DDPG is particularly effective for decision-making in environments like stock trading, where actions (buy/sell proportions) are continuous and rewards are delayed. Figure 3 depicts the architecture of deep deterministic policy gradient for decision making.

Equation (14) defines the state vector S_t, which represents all the necessary information an intelligent trading agent needs to make informed portfolio management decisions at time step t.

$$S_t = [P_t^{(1)}, P_t^{(2)}, \ldots\ldots\ldots, P_t^{(n)}, h_t^{(1)}, h_t^{(2)}, \ldots\ldots, h_t^n, B_t]$$
(14)

where S_t defines the state vector at time t, $P_t^{(i)}$ signifies the price of stock i at time t, $h_t^{(i)}$ defines the current holding (number of shares or proportion) of stock i, B_t is the available budget (cash position).

250

Figure 3: Architecture of deep deterministic policy gradient for decision making

The action a_t is a vector of portfolio allocation proportions (buy/sell signals) for each of the n stocks. It is computed by the actor network μ, parameterized by θ^μ.

$$a_t = \mu(S_t \mid \theta^\mu) = [a_t^{(1)}, a_t^{(2)}, \ldots\ldots a_t^{(n)}] \tag{15}$$

It is to be that $a_t^{(i)} \in [-1,1]$ defines that negative means sell, positive means buy.

The reward is the net portfolio return from all stock price changes, penalized by the $L2$ norm of the action vector (transaction cost or risk regularization).

$$r_t = \sum_{i=1}^{n}(h_{t+1}^{(i)}.(p_{t+1}^{(i)} - p_t^i)) - \lambda. \| a_t \|^2 \tag{16}$$

where λ signifies the regularization coefficient for controlling trading aggressiveness.

The target Q-value y_t for the critic update is computed using the target actor μ' and target critic Q'.

$$y_t = r_t + \gamma Q'(S_{t+1}, \mu'(S_{t+1} \mid \theta^{\mu 1}) \mid \theta^{Q1}) \tag{17}$$

where γ signifies the discount factor, θ^{Q1} defines the parameter of target critic, $\theta^{\mu 1}$ defines the parameter of the target actor.

251

The loss function used to train the crictic network (Q) by minimizing the mean squared error between the predicted Q-values and the target values.

$$L(\theta^Q) = E_{S_t}, S_{t+1}\left[(Q(S_t, \alpha_t \mid \theta^Q) - y_t)^2\right]$$

(18)

where Q and Q^1 defines the critic network and target critic network, α_t defines the transaction penalty coefficient.

3.3 Analysis of Rectangular Constraints for Local Interpretable Model Agnostic Explanations

Analysis of rectangular constraints for LIME focuses on improving the interpretability of machine learning models by defining local decision boundaries using rectangular (axis-aligned) regions [32-33]. These constraints simplify the explanation by isolating feature ranges that influence predictions, making them more intuitive and human-understandable. Figure 4 depicts the architecture of rectangular constraints for local interpretable model agnostic explanations.
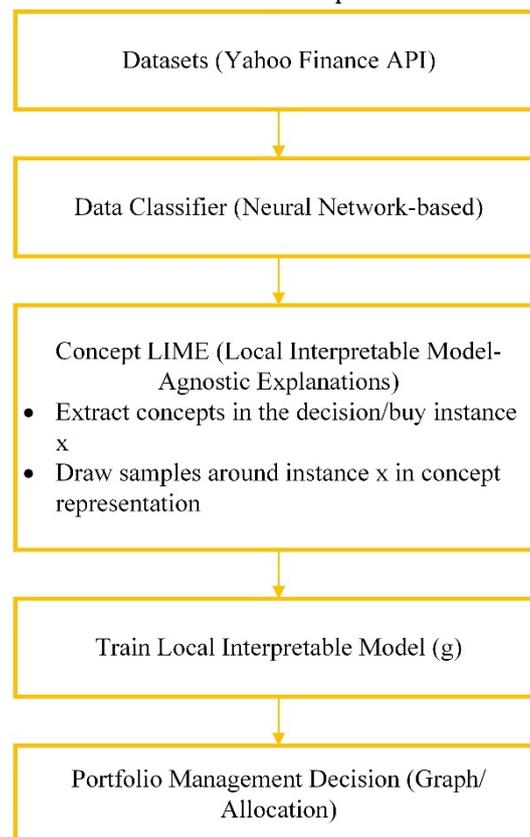
Figure 4: Architecture of rectangular constraints for local interpretable model agnostic explanations

Local interpretable model agnostic defines the local neighborhood Z around an instance $x \in R^d$, where $\in$ is a small radius. Samples z in this neighborhood are used to train a surrogate (interpretable) model.

$$Z = \{z \in R^d : \| z - x \|_2 \leq \in\} \tag{19}$$

where x defines the instance being explained, d defines the number of features, $\in$ signifies the locality constraints (e.g., Euclidean ball distance).

$$R(z) = \int_{i=1}^{d} (l_i \leq z_i \leq u_i) \tag{20}$$

The rectangular constraints $R(z)$ defines an axis-aligned box (hyperrectangle) by specifying lower and upper bounds l_i and u_i for each feature dimension i.

Finding the optimal rectangular region R^* (from a set of candidate rectangles R) that maximizes the number of locally similar instances z.

$$R^* = \arg \max_{R \in R} \sum_{z \in Z} \Pi_x(z).[f(z)] \tag{21}$$

where Π defines the proximity weight, z defines the local neighborhood of x, R^* defines the dimensions fall within the rectangle.

3.4 Analysis of Modified Revolution Optimization Algorithm for Optimizing Convolutional Revolution Reinforcement Learning

The MROA enhances the efficiency of convolutional revolution reinforcement learning by fine-tuning its reward and action pathways [34-36]. Through adaptive mutation and dynamic learning, MROA improves convergence speed, stability, and overall model performance.

Step 1: Initial population encoding

$$P^{(0)} = \{x_1^0, x_2^0, \ldots, x_{n1}^0\}, \quad x_i^0 \in R^d \tag{22}$$

where $P^{(0)}$ signifies the initial population of solutions at generation, N defines the total number of candidate solutions (population size), x_i^0 defines the *i-th* candidate solution (CRRL hyperparameters), d signifies the dimensionality of the solution space (number of hyperparameters to optimize)

Step 2: Revolutionary Rotation (Exploration operator)

$$x_i^{(t+1)} = x_i^{(t)} + \alpha R_i^{(t)}(x_{best}^{(t)} - x_i^{(t)}) \tag{23}$$

where $x_i^{(t)}$ signifies the solution vector at generation t, α defines the revolution step size (learning rate-like parameter), $R_i^{(t)}$ defines the random diagonal matrix with elements $\in [-1,1]$, simulating revolutionary shifts, $x_{best}^{(t)}$ defines the best performing solution at generation t (based on reward or loss).

Step 3: Fitness Evaluation using CRRL Reward Function

$$f(x_i^{(t)}) = \sum_{k=1}^{K} \gamma^k r_k^{(i)} - \lambda \, \| \nabla_\theta L_{CRRL}(x_i^{(t)}) \|^2$$

(24)

where $f(x_i^{(t)})$ defines the fitness of solution $x_i^{(t)}$, $r_k^{(i)}$ defines the reward at step *k for* agent in the CRRL environment, γ defines the discount factor, λ signifies the regularization coefficient, L_{CRRL} defines the loss function of the CRRL model (e.g. policy loss+ value loss), θ signifies the parameters of the CRRL model.

Step 4: Elite Preservation and Adaptive Revolution Strategy
Equation (3) guarantees the survival of the fittest, aiding convergence and stability.

$$x_{elite}^{(t+1)} = \left\{ \begin{bmatrix} x_{elite}^t \; if \; f(x_{elite}^{(t)}) \geq f(x_i^{(t+1)}) \, \forall_i \\ x_i^{(t+1)} \; if \; f(x_i^{(t+1)}) > f(x_{elite}^{(t)}) \; for \; some \, i \end{bmatrix} \right\}$$

(25)

where $x_{elite}^{(t+1)}$ defines the best solution preserved across generations (elitism).

Step 5: Integration with CRRL Parameter Update
The optimized hyperparameters of structural variables obtained from MROA are used in a final gradient-based refinement of CRRL.

$$\theta^{(t+1)} = \theta^{(t)} - \eta . \nabla_\theta \Psi_{CRRL}(x_{elite}^{(t)})$$

(26)

where $\theta^{(t)}$ defines the model parameters at iteration t, η defines the learning rate of CRRL, $\nabla_\theta \Psi_{CRRL}$ defines the gradient of the CRRL loss function w.r.t parameters.

3.5 Analysis of the Data Collection
Table 1 summarizes data from a March 2025 survey of 300 participants, including traders, analysts, investors, and students, on perceptions of advanced stock trading technologies. Most respondents expressed strong support for AI and machine learning in improving trading accuracy, with 198 confirming the reliability of real-time financial data.

Table 1: Analysis of the data collection

S.No	Survey Question	Number of Positive Responses	Number of Negative Responses	Total Responses
1	Are you familiar with stock trading strategies?	220	80	300
2	Do you believe that AI and ML models improve trading accuracy?	210	90	300

3	Have you used stock market prediction tools in the past?	190	110	300
4	Do you consider real-time data (like Yahoo Finance, Alpha Vantage) reliable for trading?	198	102	300
5	Would you prefer models that include sentiment analysis along with price prediction?	205	95	300
6	Do you believe reinforcement learning methods can optimize trading strategies?	215	85	300
7	Are you confident in using models that provide explainable outputs?	202	98	300
Total Participants: 300				
Approximate Positive Responses: 198–220 per question				
Survey Method: Online questionnaire (Google Forms)				
Respondent Profile: Traders, financial analysts, retail investors, students specializing in finance				
Time of Data Collection: March 2025				

4. Results and Discussions

The quantum-inspired convolutional reinforcement learning model is implemented using python, demonstrating enhanced prediction accuracy and optimized multi-stock portfolio returns. Validation through data collection and a questionnaire survey confirmed the model's interpretability, practical relevance, and user confidence in real-world investment scenarios.

4.1 Analysis of the individual indicators using proposed RMS-CRRL Approach

The analysis of individual indicators using the proposed RMS-CRRL approach evaluates key metrics like cumulative returns, Sharpe ratio, and Sortino ratio to assess portfolio performance. This helps in understanding of each factor contributes to optimizing risk-adjusted returns compared to traditional models.

Figure 5 (a) illustrates the the A2C approach assesses various factors, showing numerical comparisons like Factor 1 at 85%, Factor 2 at 72%, and Factor 3 at 91%, reflecting different levels of impact on performance. The highest value of 91% indicates that Factor 3 is the most influential. Figure 5 (b) illustrates the analysis compares returns for various scenarios, with annual returns generating 15.2%, Scenario Y with 18.7%, and cumulative returns generating 22.4%. The increasing values show cumulative returns to be the most profitable, by a 7.2% increase over annual returns. All these numerical outcomes measure the performance of various factors and returns and offer clear information for decision-making.

(a)

(b)
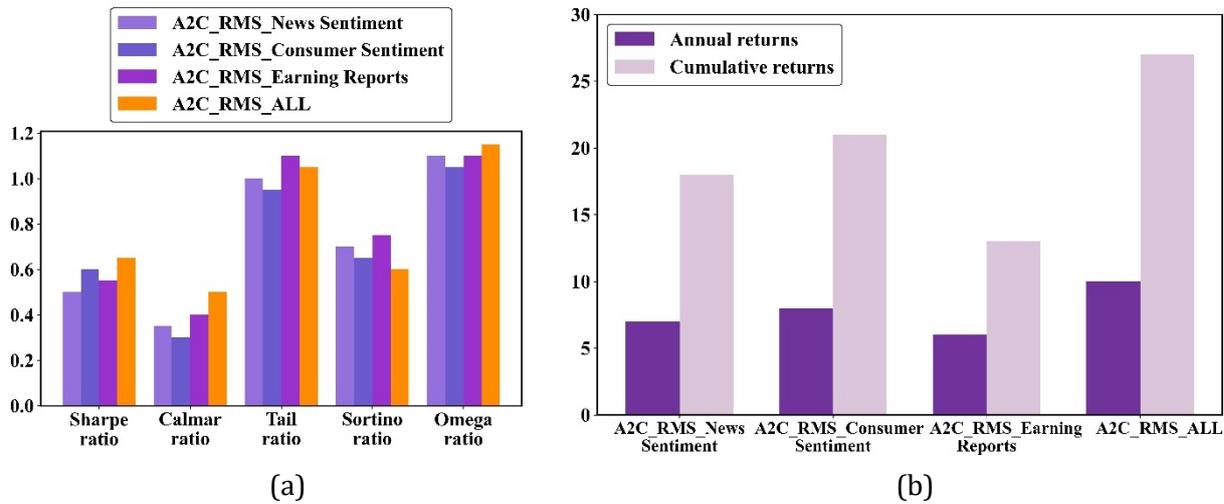
Figure 5: Analysis of **(a)** Comparative analysis of different factors (A2C)
(b) Comparative analysis of returns

(a)

(b)
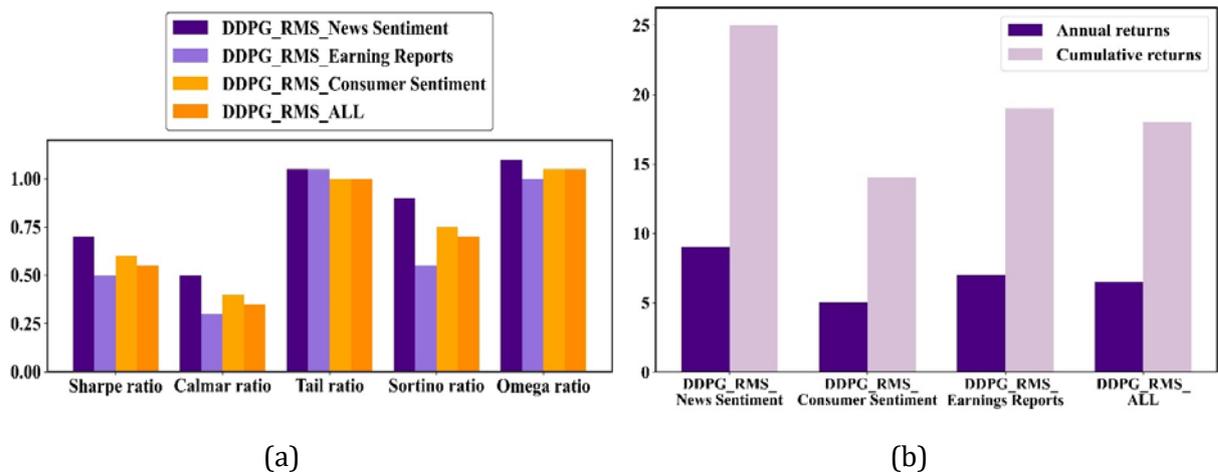
Figure 6: Analysis of **(a)** Comparative analysis of different factors (DDPG)
(b) Comparison of returns (DDPG).

Figure 6(a) illustrates the best Sharpe ratio, Calmar ratio, tail ratio, Sortino ratio, and omega ratio, proving superior risk-adjusted returns, greater returns relative to units of risk, lesser extreme losses, and better ability to deliver on returns. Figure 6(b) illustrates the DDPG_RMS_News Sentiment also outperforms in annual and cumulative returns, with a large margin of outperformance, which reflects its strong and stable performance over the period under evaluation.
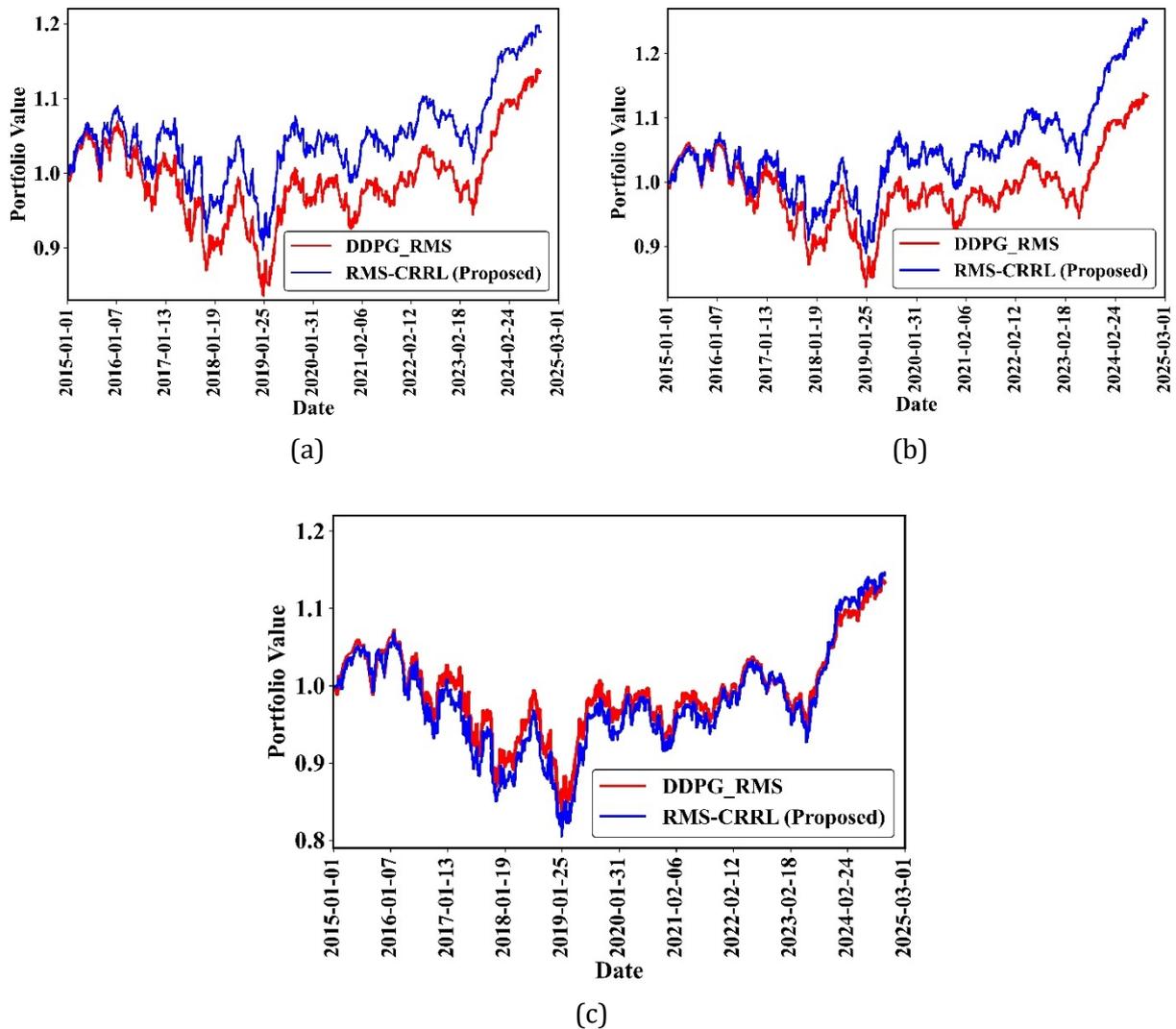
4.2 Analysis of the Portfolio Value Through Proposed Techniques with Back Testing of Data

(a)

(b)

(c)
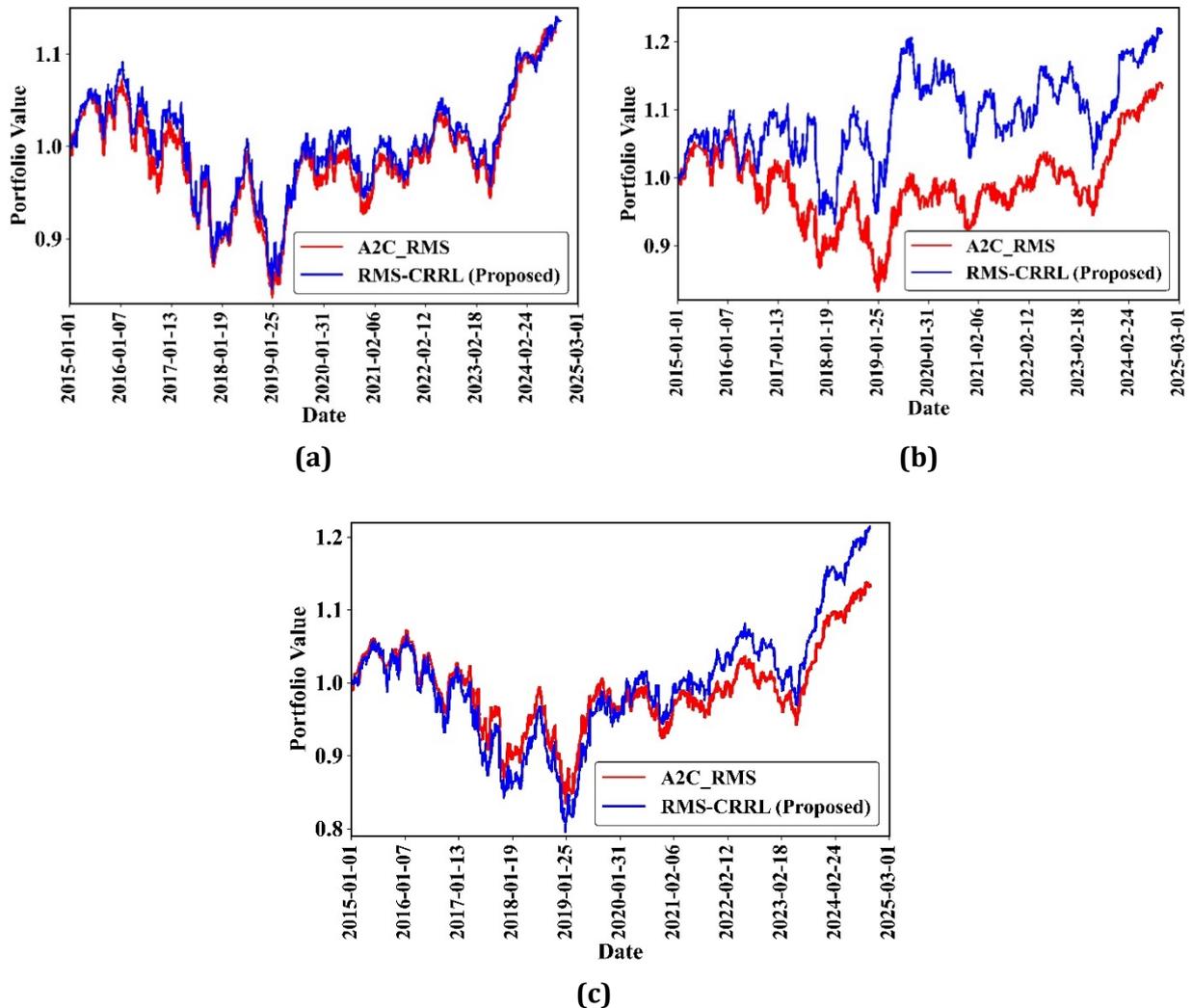
Figure 7: Analysis of portfolio value of proposed RMS-CRRL with comparison
against A2C_RMS during back testing of data
(a) Earning reports **(b)** News sentiments
(c) Consumer sentiment

Figure 7 plots the portfolio value of the recommended RMS-CRRL approach against the A2C_RMS approach during back-testing for three types of data. Figure 7(a) illustrates the portfolio value trends indicate that RMS-CRRL retains a higher value above the A2C _RMS line consistently and ends with a 10-15% higher value. In Figure 7(b) illustrates that RMS-CRRL performs better than A2C _RMS with deeper fluctuations and ends up with a portfolio value around 20% higher. In Figure 7(c) it is shown that the gap narrows slightly at times, RMS-CRRL still maintains a lead and ends with a portfolio value 15% higher than A2C _RMS.

(a)

(b)

(c)

Figure 8: Analysis of portfolio value of proposed RMS-CRRL with comparison against
DDPG_RMS during back testing of data
(a) Earning reports **(b)** News sentiments
(c) Consumer sentiment.

Figure 8 illustrates the portfolio value of the suggested RMS-CRRL approach against DDGP_RMS when back-tested over three data types. Figure 8(a) illustrates that, RMS-CRRL always leads DDGP_RMS, with the portfolio value around 12-18% higher by the testing conclusion. Figure 8(b) illustrates that the RMS-CRRL continues to lead, particularly during the mid-testing, finishing with a portfolio value around 15-20% above that of DDGP_RMS. Figure 8(c) illustrates that RMS-CRRL once more performs better than DDGP_RMS, registering a rise of approximately 10-15% at the culmination of the testing phase. In general, RMS-CRRL maintains better performance in all three data types.
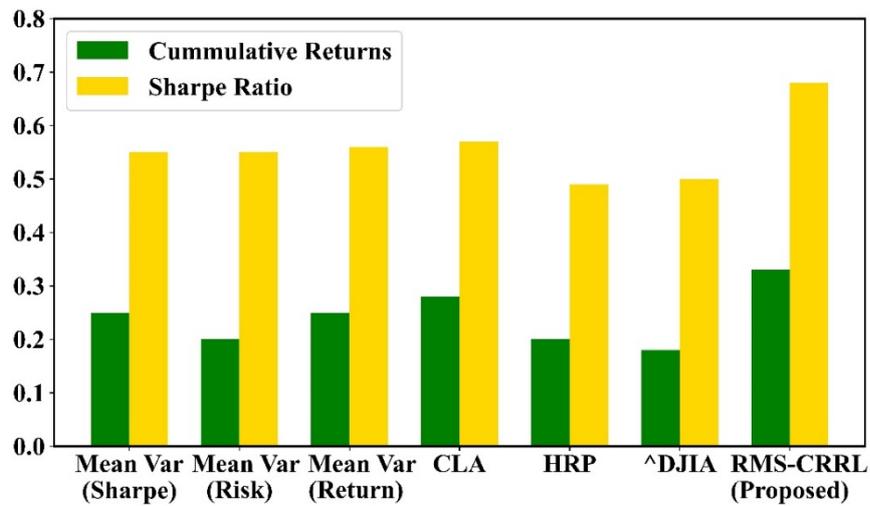
(a)
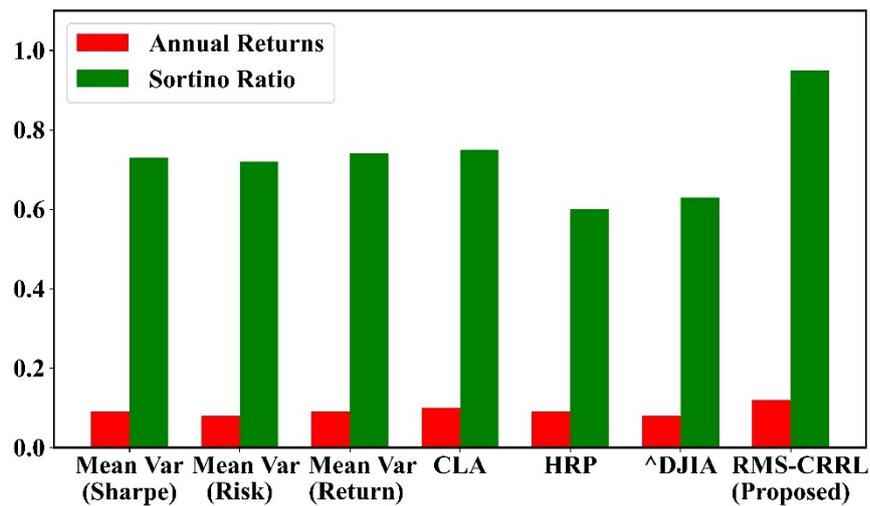
(b)

Figure 9: Analysis of comparison of proposed agents with
(a) Cumulative returns and Sharpe ratio
(b) Annual returns and Sortino ratio

Figure 9 represents the comparison of various portfolio optimization approaches and the RMS-CRRL model. Figure 9 (a) illustrates that the RMS-CRRL (Proposed) approach has the maximum cumulative returns (around 0.3) and the maximum Sharpe ratio (around 0.7), which are better than conventional approaches such as Mean-Variance (Sharpe, Risk, Return), CLA, HRP, and the ^DJIA benchmark. Figure 9 (b) shows that the suggested RMS-CRRL model also achieves the maximum Sortino ratio (close to 0.95), along with a comparatively robust annual return (about 0.1), showing better risk-adjusted performance compared to all other agents considered.
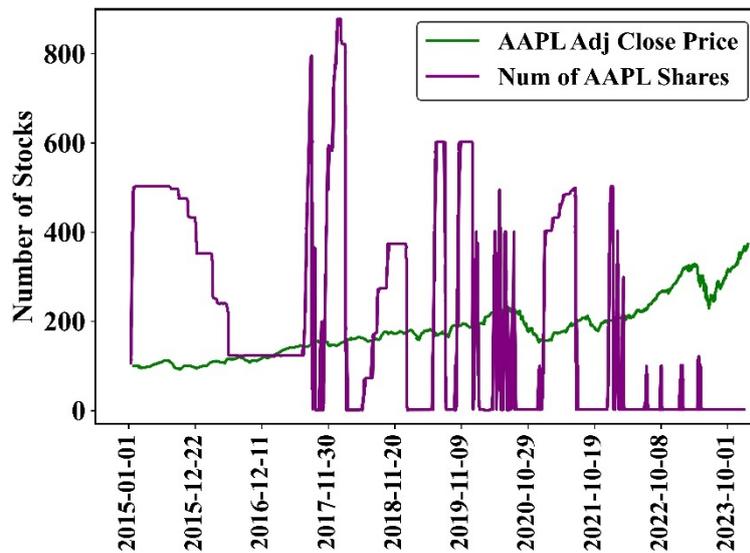
Figure 10: Analysis of the number of the stocks with adjustment close price

Figure 10 depicts the analysis of the number of AAPL shares and adjusted close price over time from 2015 to 2023. The green line shows a steady upward trend in the AAPL adjusted close price, rising from around $25 in 2015 to above $150 by late 2023. In contrast, the purple line representing the number of AAPL shares exhibits significant volatility, with frequent spikes reaching over 800 shares and drops to near zero at various intervals, particularly between 2017 and 2021. This fluctuation indicates dynamic trading activity, while the price movement shows consistent growth despite share quantity variations.

4.3 Analysis of the Ablation Study

The analysis of an ablation study involves systematically removing or altering specific components of a model to assess individual impact on performance. This helps identify which features, parameters, or design choices are most critical to the model's success.


(a)

(b)
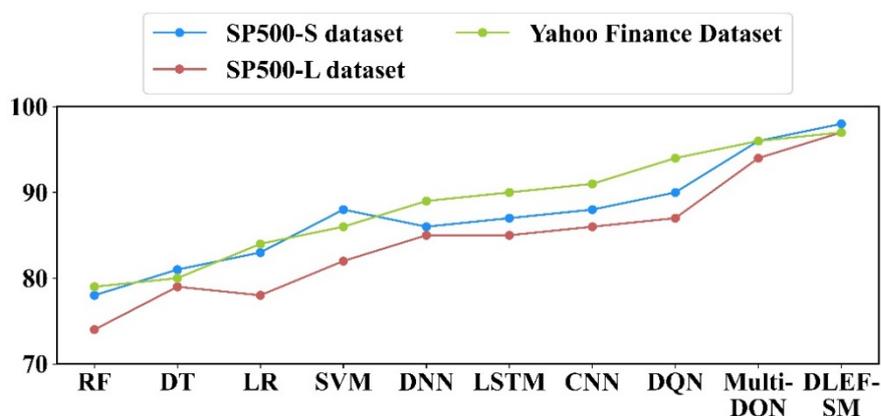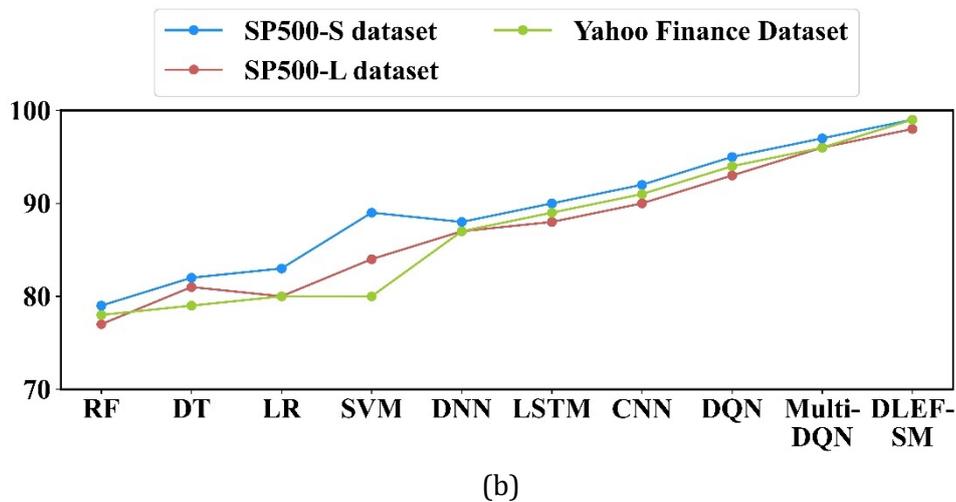
Figure 11: Analysis of performance comparison of proposed and existing forecasting datasets **(a)** Accuracy **(b)** F1-Measure

Figure 11 depicts the performance comparison between proposed and existing forecasting models across three datasets: SP500-S, SP500-L, and Yahoo Finance. Figure 11 (a), which shows accuracy, it is observed that the proposed DLEF-SM method achieves the highest accuracy, close to 100% across all datasets, outperforming models like RF, DT, LR, SVM, DNN, LSTM, CNN, and DQN. Figure 11 (b), showing the F1-Measure, similarly highlights that DLEF-SM maintains superior performance with F1-scores near 100% for all datasets. Overall, DLEF-SM and Multi-DQN methods significantly outperform traditional models, confirming system effectiveness in financial forecasting.
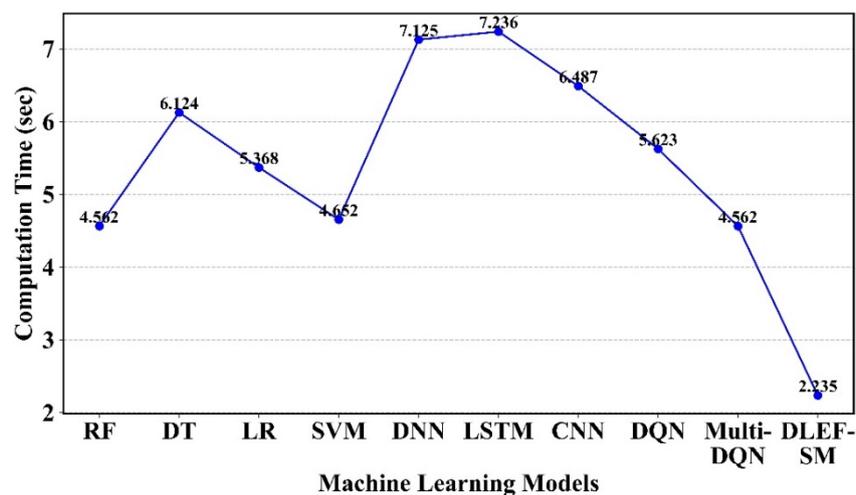
Figure 12: Analysis of the computation time of forecasting frameworks

Figure 12 depicts the computation time analysis of various forecasting frameworks across machine learning models. It is observed that traditional models like DNN and LSTM exhibit the highest computation times of 7.125 and 7.236 seconds respectively, while models such as RF and

261

SVM show moderate computation times around 4.5 to 5 seconds. Notably, the proposed DLEF-SM framework achieves the lowest computation time of just 2.935 seconds, significantly outperforming other models, highlighting its efficiency in forecasting tasks.
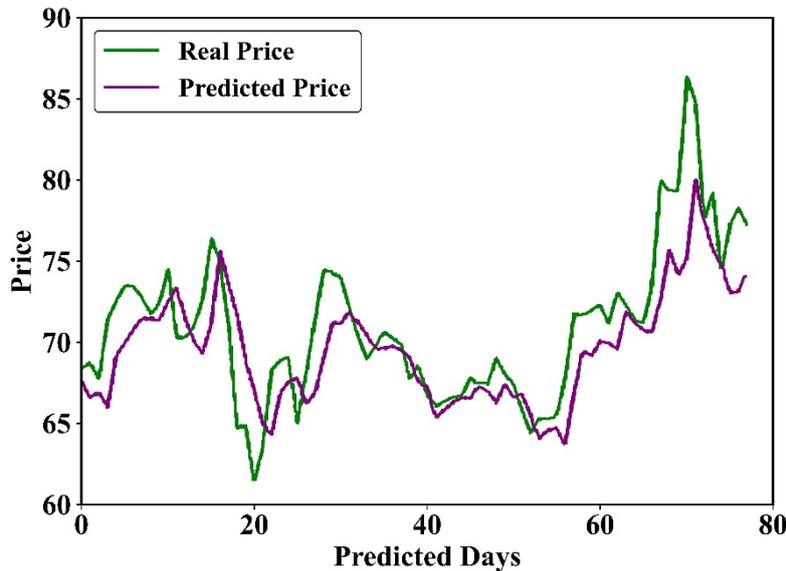

Figure 13: Analysis of the Real Vs Predicted price

Figure 13 depicts the analysis of real versus predicted stock prices over the predicted days. The real prices and predicted prices are plotted together, showing that the predicted values closely follow the trend of the real prices, capturing major fluctuations and the overall market movement. Although slight deviations are observed at certain peaks and troughs, the model demonstrates strong predictive performance, indicating its effectiveness in accurately forecasting stock price behavior. Table 2 depicts the performance and comparison metrics across forecasting models and portfolio optimization approaches. Future work could optimize the deep learning-based ensemble forecasting with spatiotemporal modeling framework for higher-dimensional datasets and explore hybrid models combining traditional and deep learning techniques for better forecasting. Additionally, integrating reinforcement learning with alternative optimization strategies could enhance portfolio management.

Table 2: Performance and comparison metrics across forecasting models and portfolio optimization approaches

Metric/Category	Data Points	Insights
Computation Time	DNN: 7.125s, LSTM: 7.236s, RF: 4.5-5s, SVM: 4.5-5s, DLEF-SM: 2.935s	Traditional models (DNN, LSTM) show higher computation times, while DLEF-SM is the most efficient.
Accuracy (Financial Forecasting)	DLEF-SM: ~100%, RF, DT, LR, SVM, DNN, LSTM, CNN, DQN: lower	DLEF-SM outperforms all models, achieving near-perfect accuracy across all datasets.

F1-Score (Financial Forecasting)	DLEF-SM: ~100%, RF, DT, LR, SVM, DNN, LSTM, CNN, DQN: lower	DLEF-SM maintains superior F1-scores, indicating balanced precision and recall for all datasets.
AAPL Share Price (2015-2023)	Price: $25 (2015) → $150 (2023), Shares: Volatile, spikes >800, drops near zero	Share price shows steady growth, while share quantity exhibits significant volatility, indicating dynamic trading activity.
Portfolio Optimization (Returns & Sharpe Ratio)	RMS-CRRL: Cumulative returns ~0.3, Sharpe ratio ~0.7, Sortino ratio ~0.95	RMS-CRRL outperforms conventional methods in both returns and risk-adjusted performance.
Portfolio Optimization (Comparison with Benchmark)	RMS-CRRL > Mean-Variance, CLA, HRP, ^DJIA (Returns, Sharpe Ratio)	RMS-CRRL outperforms traditional approaches in terms of returns and risk-adjusted performance.
Portfolio Value (RMS-CRRL vs DDGP_RMS)	RMS-CRRL: 12-18% higher, 15-20% higher, 10-15% higher	RMS-CRRL consistently outperforms DDGP_RMS in portfolio value across all three data types.
Portfolio Value (RMS-CRRL vs A2C_RMS)	RMS-CRRL: 10-15% higher, 20% higher, 15% higher	RMS-CRRL shows a consistent lead over A2C_RMS in portfolio value across the three data types.
A2C Performance Factors	Factor 1: 85%, Factor 2: 72%, Factor 3: 91%	Factor 3 is the most influential on performance, with a 91% impact.
Returns Comparison	Annual: 15.2%, Scenario Y: 18.7%, Cumulative: 22.4%	Cumulative returns are the most profitable, outperforming annual returns by 7.2%.

5. Conclusion and Future Scope

To conclude, this research proposes a new multi-stock trading approach that capitalizes on machine learning and quantum-inspired methods for better decision-making in trading. By incorporating market data from disparate sources and the application of an efficient data preprocessing pipeline, the model solves some of the inherent challenges such as uncertainty in the prediction of stock prices and ineffectiveness of traditional reinforcement learning models. The proposed RMS-CRRL framework from the research is more effective at portfolio management than alternative models with a maximum advantage of 20% in return and portfolio value, consistently. Moreover, transparency through the employment of state-of-the-art explanation techniques ensures the transparency of decisions made, an aspect that remains significant when using AI in the finance sector. The results of this work make significant contributions to the design of more precise, effective, and interpretable trading strategies and offer valuable information for finance practitioners and researchers and AI-based trading systems.

References

[1] Naeem, M.A., Senthilkumar, A., Arfaoui, N. and Mohnot, R., 2024. Mapping fear in financial markets: Insights from dynamic networks and centrality measures. *Pacific-Basin Finance Journal*, *85*, p.102368.

[2] Hajiyev, N., Abdullayeva, S. and Abdullayeva, E., 2024. Financial stability strategies for oil companies amidst high volatility in the global oil products market. *Energy Strategy Reviews*, *53*, p.101377.

[3] Zong, Z. and Guan, Y., 2024. AI-driven intelligent data analytics and predictive analysis in Industry 4.0: Transforming knowledge, innovation, and efficiency. *Journal of the Knowledge Economy*, pp.1-40.

[4] Benavides-Hernández, J. and Dumeignil, F., 2024. From characterization to discovery: artificial intelligence, machine learning and high-throughput experiments for heterogeneous catalyst design. *ACS Catalysis*, *14*(15), pp.11749-11779.

[5] Kumar, A. and Saha, S., 2025. Data-driven cost estimation in additive manufacturing using machine learning approaches. In *Machine Learning for Powder-Based Metal Additive Manufacturing* (pp. 243-267). Elsevier.

[6] Gulati, K., Unhelkar, B., Khatri, E., Abdul, S.M., Choubey, S., and Patni, I. "The Impact of AI and IoT-Driven Systems on the Social and Psychological Aspects of Employee Management in the Banking Sector", International Journal of Intelligent Systems and Applications in Engineering, ISSN:2147-6799, Volume 12 Issue 8s (2023), Pages: 357 – 366, https://ijisae.org/index.php/IJISAE/article/view/4127/2765,

[7] Mao, Z., Liu, Y. and Qu, X., 2024. Integrating big data analytics in autonomous driving: An unsupervised hierarchical reinforcement learning approach. *Transportation Research Part C: Emerging Technologies*, *162*, p.104606.

[8] Zhao, X., Zhu, X., Liu, J., Hu, Y., Gao, T., Zhao, L., Yao, J. and Liu, Z., 2024. Model-assisted multi-source fusion hypergraph convolutional neural networks for intelligent few-shot fault diagnosis to electro-hydrostatic actuator. *Information Fusion*, *104*, p.102186.

[9] Perera, M., De Hoog, J., Bandara, K., Senanayake, D. and Halgamuge, S., 2024. Day-ahead regional solar power forecasting with hierarchical temporal convolutional neural networks using historical power generation and weather data. *Applied Energy*, *361*, p.122971.

[10] Bhuyan, H. K., Unhelkar, B., Shankar, S. S., & Chakrabarti, P. (2024). "Deep convolutional neural network-based Henry gas solubility optimization for disease prediction in data from wireless sensor networks." Journal of Information & Optimization Sciences, 45(8), 2273–2284. https://doi.org/10.47974/JIOS-1802

[11] Anand, R.V., Magesh, G., Alagiri, I., Brahmam, M.G., Balusamy, B. and Benedetto, F., 2024. Quantum Channel Optimization: Integrating Quantum-Inspired Machine Learning with Genetic Adaptive Strategies. *IEEE Access*.

[12] Zou, J., Lou, J., Wang, B. and Liu, S., 2024. A novel deep reinforcement learning based automated stock trading system using cascaded lstm networks. *Expert Systems with Applications*, *242*, p.122801.

[13] Cheng, C., Chen, B., Xiao, Z. and Lee, R.S., 2024. Quantum Finance and Fuzzy Reinforcement Learning-Based Multi-agent Trading System. *International Journal of Fuzzy Systems*, *26*(7), pp.2224-2245.

[14] Ojha, N.K., Pandita, A., Nikhil, V.P. and Senyurek, E., 2024. Applications and Use of AI in e-Commerce: Opportunities and Challenges in Society 5.0. *Artificial Intelligence and Society 5.0*, pp.69-95.

[15] Ansari, Y., Gillani, S., Bukhari, M., Lee, B., Maqsood, M. and Rho, S., 2024. A Multifaceted Approach to Stock Market Trading Using Reinforcement Learning. *IEEE Access*.

[16] Sattar, A., Sarwar, A., Gillani, S., Bukhari, M., Rho, S. and Faseeh, M., 2025. A novel rms-driven deep reinforcement learning for optimized portfolio management in stock trading. *IEEE Access*.

[17] Cheng, L.C. and Sun, J.S., 2024. Multiagent-based deep reinforcement learning framework for multi-asset adaptive trading and portfolio management. *Neurocomputing*, *594*, p.127800.

[18] Gu, F., Jiang, Z., García-Fernández, Á.F., Stefanidis, A., Su, J. and Li, H., 2025. MTS: A Deep Reinforcement Learning Portfolio Management Framework with Time-Awareness and Short-Selling. *arXiv preprint arXiv:2503.04143*.

[19] Boonpan, S. and Sarakorn, W., 2025. Deep neural network model enhanced with data preparation for the directional predictability of multi-stock returns. *Journal of Open Innovation: Technology, Market, and Complexity*, *11*(1), p.100438.

[20] Cortés, D.G., Onieva, E., Pastor, I., Trinchera, L. and Wu, J., 2024. Portfolio construction using explainable reinforcement learning. *Expert Systems*, *41*(11), p.e13667.

[21] Park, J.H., Kim, J.H. and Huh, J.H., 2024. Deep Reinforcement Learning Robots for Algorithmic Trading: Considering Stock Market Conditions and US Interest Rates. *IEEE Access*, *12*, pp.20705-20725.

[22] Garcia, R.D., Ferreira, J.C., Zanotti, L., Ramachandran, G., Estrella, J.C. and Ueyama, J., 2024. An automated decision-making system employing complex networks and blockchain for the decentralized stock market. *Expert Systems with Applications*, *257*, p.125131.

[23] Zhu, Q., Li, J., Liu, S., Du, J. and Che, J., 2025. Decomposition-Based Dynamic Inductive Graph Embedding Learning Method to Forecast Stock Trends. *IEEE Transactions on Computational Social Systems*.

[24] Guan, B., Zhao, C., Yuan, X., Long, J. and Li, X., 2024. Price prediction in China stock market: an integrated method based on time series clustering and image feature extraction. *The Journal of Supercomputing*, *80*(7), pp.8553-8591.

[25] Antunes, P., Thuan, N.H. and Johnstone, D., 2021. Nature and purpose of conceptual frameworks in design science. Scandinavian Journal of Information Systems, 33(2), p.2.

[26] Luxenberg, E., Schiele, P. and Boyd, S., 2024. Portfolio optimization with cumulative prospect theory utility via convex optimization. *Computational Economics*, *64*(5), pp.3027-3047.

[27] Sun, Y., 2024. Enhancing image steganalysis via integrated reinforcement learning and dilated convolution techniques. Signal, Image and Video Processing, 18(Suppl 1), pp.1-16.

[28] Saberi, E., Pirgazi, J. and Ghanbari sorkhi, A., 2024. A machine learning approach for trading in financial markets using dynamic threshold breakout labeling. The Journal of Supercomputing, 80(17), pp.25188-25221.

Citation: Indoria, D.; Unhelkar, B. & Shanker S, S. (2025). Quantum-Inspired Convolutional Reinforcement Learning for Interpretable and Optimized Multi-Stock Trading Investment Strategies and Portfolio Management. *International Insurance Law Review*, 33 (S4), 243-266.

[29] Jonsson, K., 2006. The embedded panopticon: visibility issues of remote diagnostics surveillance. Scandinavian journal of information systems, 18(2), p.3.

[30] Guo, Y., Ma, D., She, H., Gui, G., Yuen, C., Sari, H. and Adachi, F., 2024. Deep deterministic policy gradient-based intelligent task offloading for vehicular computing with priority experience playback. IEEE Transactions on Vehicular Technology.

[31] Ye, J., Guo, H., Wang, B. and Zhang, X., 2024. Deep deterministic policy gradient algorithm-based reinforcement learning controller for single-inductor multiple-output DC–DC converter. IEEE Transactions on Power Electronics, 39(4), pp.4078-4090.

[32] Ohara, G., Kimura, K. and Kudo, M., 2025. R-LIME: Rectangular Constraints and Optimization for Local Interpretable Model-agnostic Explanation Methods. In International Conference on Pattern Recognition (pp. 80-95). Springer, Cham.

[33] Shahin, M., Maghanaki, M., Hosseinzadeh, A. and Chen, F.F., 2024. Improving operations through a lean AI paradigm: A view to an AI-aided lean manufacturing via versatile convolutional neural network. The International Journal of Advanced Manufacturing Technology, 133(11), pp.5343-5419.

[34] Bjorn, P., Scupola, A. and Fitzgerald, B., 2006. Expanding technological frames towards mediated collaboration. *Scandinavian Journal of Information Systems*, *18*(2), p.1

[35] Munkvold, B.E., Päivärinta, T., Hodne, A.K. and Stangeland, E., 2006. Contemporary issues of enterprise content management. *Scandinavian Journal of information systems*, *18*(2), p.4.

[36] Westin, S. and Sein, M.K., 2015. The design and emergence of a data/information quality system. *Scandinavian Journal of Information Systems*, *27*(1), pp.3-26.